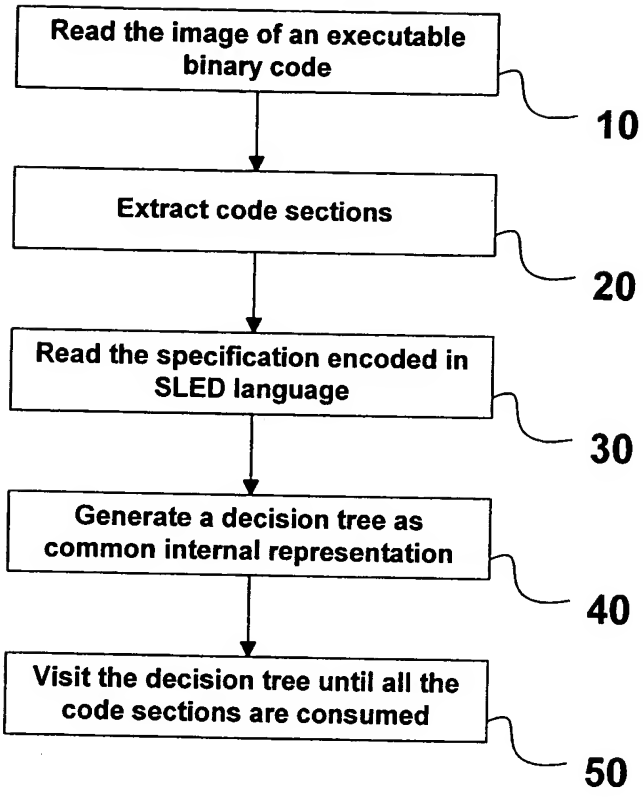
Read the image of an executable binary code
10
Extract code sections
20
Read the specification encoded in SLED language
30
Generate a decision tree as common internal representation
40
Visit the decision tree until all the code sections are consumed
50

FIG. 1

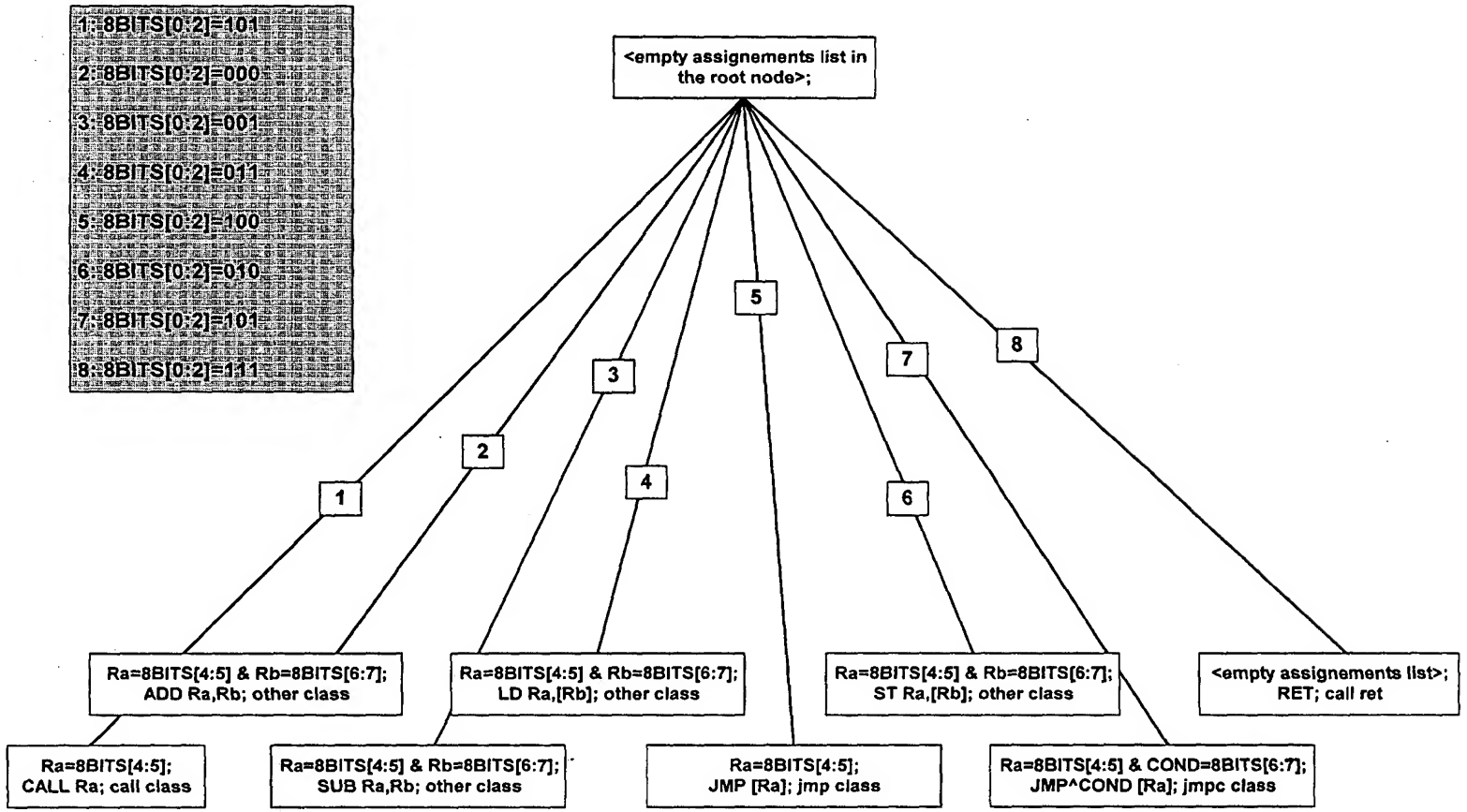
1: 8BITS[0:2]=101
2: 8BITS[0:2]=000
3: 8BITS[0:2]=001
4: 8BITS[0:2]=011
5: 8BITS[0:2]=100
6: 8BITS[0:2]=010
7: 8BITS[0:2]=101
8: 8BITS[0:2]=111
<empty assignements list in the root node>;
1
2
3
4
5
6
7
8
Ra=8BITS[4:5] & Rb=8BITS[6:7]; ADD Ra,Rb; other class
Ra=8BITS[4:5] & Rb=8BITS[6:7]; LD Ra,[Rb]; other class
Ra=8BITS[4:5] & Rb=8BITS[6:7]; ST Ra,[Rb]; other class
<empty assignements list>; RET; call ret
Ra=8BITS[4:5]; CALL Ra; call class
Ra=8BITS[4:5] & Rb=8BITS[6:7]; SUB Ra,Rb; other class
Ra=8BITS[4:5]; JMP [Ra]; jmp class
Ra=8BITS[4:5] & COND=8BITS[6:7]; JMP^COND [Ra]; jmpc class

FIG. 2

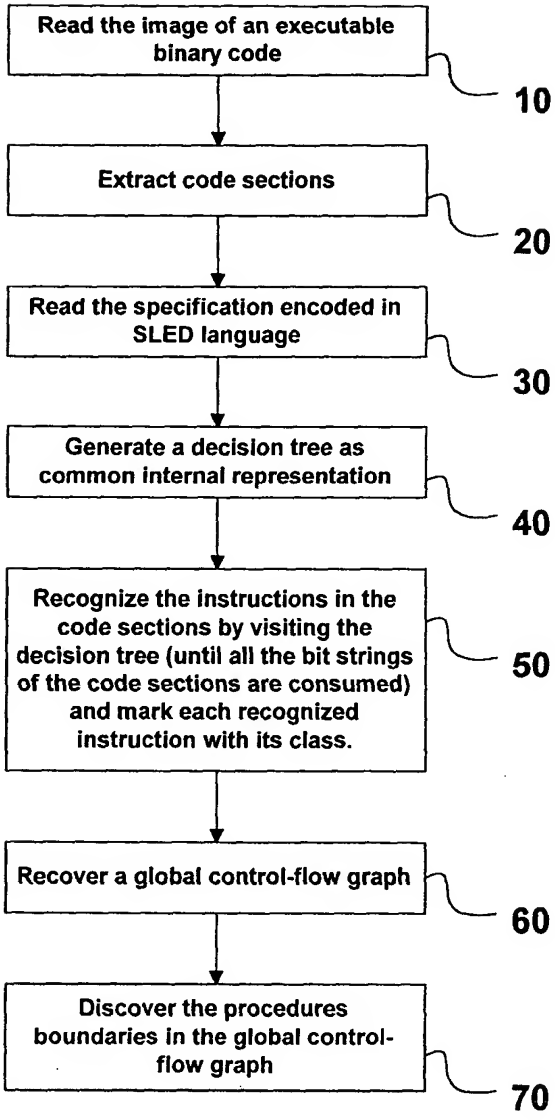

FIG. 3: flow chart for static code analyzer